

## TITLE OF THE INVENTION

Residue Calculating Unit Immune to Power Analysis

## BACKGROUND OF THE INVENTION

### Field of the Invention

5 The present invention relates to power-residue calculating units used for encryption and decryption of information applicable in the fields of telecommunications network, traffic, finance, medical services, distribution and so on. More particularly, the present invention relates to a power-residue calculating unit using a Montgomery algorithm.

### 10 Description of the Background Art

Owing to the technological development in the field of telecommunication, security (namely, to prevent criminal use or destruction of data) over a communication network has received a great deal of attention. Therefore, encryption and decryption of information are frequently used.

15 The applicable fields of encryption and decryption range from telecommunication to traffic, finance, medical services, distribution and so on. This type of encryption and decryption are required to realize an advanced security based on a simple concept.

20 To facilitate understanding of this type of technique, a concept of encryption/decryption of information will be briefly described. In the world of encryption, "asymmetric cipher algorithm" is superior. The asymmetric cipher algorithm refers to a cipher algorithm using different encryption and description keys, where one of which cannot be "easily calculated" from the other. An RSA (Rivest-Shamir-Adleman scheme) cipher using a power-residue calculation (a residue is obtained by multiplying a certain number X several times and then dividing the result by another number N) is representative of the asymmetric cipher algorithm.

25 Basically, the power-residue calculation in accordance with the following formula (1) is used to generate an RSA cipher. Formula (1) means that a residue is obtained by dividing  $X^Y$  by N. In formula (1), X represents a plaintext to be subjected to encryption (or decryption), where Y and N are keys for encryption (or decryption).

$$X^Y \bmod N \quad \dots (1)$$

The power-residue calculation facilitates encryption and decryption of information. If the operand bit lengths of X, Y, and N are increased, decryption of each key becomes difficult.

5 However, the greater operand bit length requires a longer period of time for the power-residue calculation. Then, what is most important is how to reduce the time for the power-residue calculation with a greater operand bit length.

Next, encryption and decryption processes using the power-residue calculation will be described with an RSA cipher by way of example.

10 [Encryption and Decryption of RSA Cipher]

For decryption of the RSA cipher, the following equation (2) is used.

$$C = M^e \bmod N \quad \dots (2)$$

15 For decryption, the following equation (3) is used.

$$M = C^d \bmod N \quad \dots (3)$$

20 Here, M represents a plaintext for encryption, and C represents a plaintext which has been subjected to encryption, i.e., a ciphertext. Further, e and N in equation (2) are encryption keys, whereas d and N in equation (3) are decryption keys. Further, the relationship as shown in the following equations (4) and (5) is assumed.

25  $N = p \cdot q \quad \dots (4)$

$$1 \equiv e \cdot d \bmod \{ \text{LCM} (p - 1, q - 1) \} \quad \dots (5)$$

30 Here, " $\equiv$ " indicates that the left and right sides are similar to each other, and "LCM" is an abbreviation for a least common multiple. Further, p and q are relatively prime integers. Note that e and N are public keys, whereas d, p and q are secret keys.

Equations (4) and (5) both define conditions of numeric values for the power-residue calculation in a cipher algorithm. Equation (4) indicates

that  $N$  is a product of relatively prime large prime numbers  $p$  and  $q$ . Since  $p$  and  $q$  are both odd numbers,  $N$  should also be an odd number. Equation (5) indicates that a residue, which is obtained by dividing a product of  $e$  and  $d$  by a least common multiple of values obtained by subtracting 1 respectively from  $p$  and  $q$  shown in equation (4), is 1.

Under the conditions specified in equations (4) and (5), plaintext  $M$  is encrypted by equation (2), and encrypted plaintext  $M$  (ciphertext  $C$ ) is decrypted by equation (3).

[Method of Power-Residue Calculation]

10 A method of a power-residue calculation used for encryption/decryption will now be described. The power-residue calculation for  $A = M^e \bmod N$  is carried out with use of an iterative square product method as shown in the following flow 1 with a binary digit expansion of an integer  $e$  being  $e^{k-1} \dots e^1 e^0$ .

15 (Flow 1)

begin

$A = 1$

    for  $i = k - 1$  to 0

        begin

$A = A^2 \bmod N$

... (6)

            If  $e^i = 1$  then  $A = A \cdot M \bmod N$

... (7)

        end

    end

25 A solution of the power-residue calculation would be equal to  $A$ .

As described above, the calculation is based on multiplication and division (mod calculation) as shown in equations (6) and (7). The multiplication provides  $A \times A$  or  $A \times M$  for a value of  $A$  having an initial value of 1. The division provides  $\bmod N$  for a value obtained by each multiplication (a calculation of a residue when divided by  $N$ ). Calculations are iteratively performed in accordance with a bit value of "e" with a pair of "multiplication and division" ( $A \times A \bmod N$ ,  $A \times M \bmod N$ ). More specifically, "multiplication and division" is performed in accordance with

each bit starting from the most significant bit to the least significant bit of "e".

As described above, in the power-residue calculation, a solution is obtained by iteratively performing basic residual calculations (mod calculation). The iterative frequency per se is at most several hundreds to several thousands of times, which can be processed by a software at high speed.

However, to carry out the residue calculation per se, i.e., division, by a hardware, a large calculation circuit and a complicated process are required, and there have been demand for an improvement. Since large integers of about 1024 bits are usually used for e, d, M, N and so on, even a high-speed exponential calculation still requires multiple precision multiplication and residual calculation of about 1500 times on average per RSA calculation. In particular, various high-speed methods, including an approximation method, residual table method and Montgomery algorithm, have been proposed for the residual calculation.

To increase the speed of the power-residue calculation mostly used for a public key cryptograph representative of the RSA cipher, the speed of one residual calculation must be increased. A Montgomery algorithm provides high speed residual calculation. In particular, in the multiplication residual calculation, division can be simplified by e.g., bit shift. Thus, the power-residue calculation used for a public key cryptograph (e.g., RSA cipher) can be performed at higher speed.

On the other hand, the Chinese remainder theorem states that a calculation modulo a composite number can be carried out by calculations modulo relatively prime factors of the composite number. If this is applied to RSA encryption with 1024-bit length, in practice, only a calculating circuit with a modulus of an integer of a 512-bit length (here corresponding to p and q), rather than a power-residue calculating circuit modulo N of a 1024-bit length, is required as hardware. This contributes to miniaturization of the hardware.

As described above, the size of the calculating circuit disadvantageously increases since the power-residue calculation involves a

highly complicated process of basic residue calculation (mod calculation). Then, Montgomery has proposed that a solution can be obtained by "multiplication" and a simple bit-string process, rather than by the above described general method of residual calculation (mod calculation). The 5 method proposed by Montgomery will be briefly described in the following.

[Montgomery Algorithm]

A Montgomery algorithm implementing high speed residual calculation will be described.

The Montgomery algorithm is based on the fact that use of residual 10 modulus  $N$  ( $N > 1$ ) and a cardinal number  $R$  ( $R > N$ ) which is relatively prime with respect to residual modulus  $N$  allows calculation of  $TR^{-1} \bmod N$  to be performed only by division by cardinal number  $R$  with a dividend being  $T$ . This eliminates the need for division by  $N$  for the residual calculation.

Here,  $N$ ,  $R$ ,  $R^{-1}$  and  $T$  are integers. Dividend  $T$  satisfies  $0 \leq T < R \cdot N$ . 15  $R^{-1}$  is an inverse of cardinal number  $R$  according to residual modulus  $N$ . Further, consider an integer  $N'$  that satisfies a relation of  $R \cdot R^{-1} - N \cdot N' = 1$  ( $0 \leq R^{-1} < N$ ,  $0 \leq N' < R$ ). Further, if a power of 2 is used for cardinal number  $R$ , the division by cardinal number  $R$  can be replaced by a shift operation. Thus, a high speed calculation of  $T \rightarrow TR^{-1} \bmod N$  ( $TR^{-1} \bmod N$  with a 20 dividend being  $T$ ) is enabled.

An algorithm  $MR(T)$  of  $T \rightarrow TR^{-1} \bmod N$  is given below as algorithm 1. Note that in algorithm 1,  $(T+m \cdot N)/R$  has been proved to be always divisible.

(Algorithm 1) Algorithm  $Y = MR(T)$  of  $T \rightarrow TR^{-1} \bmod N$  is given by 25 the following sequence.

$$M = (T \bmod R) \cdot N' \bmod R \quad \dots (8)$$

$$Y = (T+m \cdot N)/R \quad \dots (9)$$

if  $Y \geq N$  then  $Y = Y - N$

30  $Y < N$  then return  $Y$

A single  $MR$  provides only  $TR^{-1} \bmod N$  rather than a residue  $T \bmod N$ . Thus, to find residue  $T \bmod N$ , an  $MR$  calculation is again performed

using a product of MR (T) and preliminary found  $R^2 \bmod N$  as shown below.

$$\begin{aligned} & \text{MR (MR (T) \cdot (R}^2 \bmod N)) \\ & = (\text{TR}^{-1} \bmod N) \cdot (\text{R}^2 \bmod N) \cdot \text{R}^{-1} \bmod N \\ 5 & = \text{TR}^{-1} \cdot \text{R}^2 \cdot \text{R}^{-1} \bmod N \\ & = \text{T mod N} \end{aligned}$$

Thus, residue T mod N can be found.

An algorithm implementing the multiplication residue calculation  
10 by the Montgomery method using the iterative square product method  
(iterative square method) of the power-residue calculation is given below.  
Search is performed starting from an upper bit of key e and, if the bit value  
of the key is 1, the Montgomery multiplication residual calculation of MR  
(X \cdot Y) is performed.

15

$$Y = Rr \quad (Rr = R^2 \bmod N \quad (R = 2^{k+2}))$$

$$X = M$$

$$X = \text{MR (X} \cdot Y) \quad \dots (10)$$

$$Y = \text{MR (1} \cdot Y) \quad \dots (11)$$

20

for j = k to 1

$$\text{if } e_j == 1 \text{ then } Y = \text{MR (X} \cdot Y) \quad \dots (12)$$

$$\text{if } j > 1 \text{ then } Y = \text{MR (X} \cdot Y) \quad \dots (13)$$

end for

$$Y = \text{MR (1} \cdot Y) \quad \dots (14)$$

25

$$Y = Y \bmod N \quad \dots (15)$$

Here,  $\text{MR (X} \cdot Y)$  and  $\text{MR (Y} \cdot X)$  are equal, where  $e_j$  represents j-th bit  
of key e. In the case of an integer with 512-bit length,  $k = 512$ . The  
power-residue calculation of 512 bits can be implemented by the  
30 Montgomery multiplication residue calculation of 514 bits and the residual  
calculation of 512 bits.

The Montgomery multiplication residual calculation result  $P = \text{MR}$   
(B \cdot A) is found in the following manner by a sequential calculation of a

cardinal number  $W$  which is most suitable for being implemented as hardware.

$$W = 2^d$$

$$5 \qquad \qquad N0' = N' \bmod W$$

$$P = 0$$

for  $j = 0$  to  $k$

$$M = (P \bmod W) \cdot N0'$$

... (16)

$$P = ((P + (A \bmod W) \cdot B \cdot W + M \cdot N) / W)$$

... (17)

$$A = A/W$$

11 (18)

end

Here,  $d$  is a natural number depending on the hardware. Thus, Montgomery multiplication residual calculation result  $P$  can be found.

15 Then, 514-bit Montgomery multiplication residual calculation result  $P = MR$  ( $B \cdot A$ ) can be found in the following manner by a sequential calculation of cardinal number 2 where  $d = 1$ .

$$N0' = N' \bmod 2$$

20 P = 0

for j = 0 to 514

$$M = (P \bmod 2) \cdot NO'$$

... (19)

$$P = ((P + (A \bmod 2) \cdot B \cdot 2 + M \cdot N) / 2)$$

... (20)

$$A = A/2$$

... (21)

25 end

As described above, to implement the power-residue calculation, a common practice is to use the Montgomery method for the 512-bit length power-residue calculation in the hardware and to use a process making use of the Chinese remainder theorem in the software. There are a plurality of kinds of hardware implementations, which may be employed in practice.

As conventional techniques for the RSA cipher, for example, Japanese Patent Laying-Open No. 7-20778 discloses RSA cipher,

Montgomery method and a correction device, and Japanese Patent Laying-Open No. 11-212456 discloses RSA cipher and Montgomery method.

Furthermore, for example, Japanese Patent Laying-Open No. 10-269060 discloses RSA cipher, elliptical cryptography and Montgomery

5 method.

For hardware for encryption using the algorithm described above, a so-called "Power Analysis" scheme may be used to externally search for an encryption key.

In equipment provided with ciphers such as personal computers, 10 mobile phones and IC cards, Simple Power Analysis, Differential Power Analysis and the like are known as a scheme to observe consumed current in LSI to search for a key for encryption.

Here, the hardware for performing the encryption processing for the RSA cipher as described above desirably has a highest possible immunity 15 against Power Analysis.

For this purpose, it is desirable that the consumed current pattern of the hardware has so fewer characteristics that the internal algorithm cannot be estimated. Moreover, it is desirable that the current pattern depending on confidential data that cannot be controlled externally does not exist. In 20 addition, it is desirable that there is a low correlation between externally controllable input data and consumed current values and that there are fewer characteristics in the level pattern of the correlation between the externally controllable input data and the consumed current values.

In the power-residue calculation described above, the process is 25 performed with a bit value of "e" as a unit. Therefore it is desirable that the consumed current does not vary depending on "1" or "0" of the bit value.

#### SUMMARY OF THE INVENTION

The object of the present invention is to provide a power-residue calculating unit applicable to a cipher processing circuit and the like with a 30 reduced circuit scale, with an increased speed and with improved security against Power Analysis.

In summary, the present invention provides a power-residue calculating unit including a first register group, a second register group, a

first internal bus, a second internal bus, a Montgomery multiplication residue calculation executing portion, a power-residue calculation executing portion, and a pseudo calculation executing portion.

The first register group holds a first kind of data. The second 5 register group holds a kind of data to be referred to concurrently with the data held in the first register group. The first internal bus is connected to the first register group. The second internal bus is connected to the second register group. The Montgomery multiplication residue calculation executing portion is connected to the first and second internal buses for 10 concurrently referring to values held in the first and second register groups and executing a Montgomery multiplication residue calculation. The power-residue calculation executing portion is connected to the first and second internal buses and the Montgomery multiplication residue calculation executing portion for concurrently referring to values held in the 15 first and second register groups, communicating data with the Montgomery multiplication residue calculation executing portion, and executing a power-residue calculation. The pseudo calculation executing portion executes in a pseudo manner an intermediate calculation process that can be omitted to obtain each calculation result of the Montgomery multiplication residue calculation and the power-residue calculation. 20

Therefore the advantage of the present invention is to provide a power-residue calculating unit capable of high speed processing with a reduced circuit scale. In addition, an intermediate calculation process is executed in a pseudo manner that can be omitted to obtain the calculation 25 result of the Montgomery multiplication residue calculation and the power-residue calculation, thereby improving the immunity against Power Analysis.

The foregoing and other objects, features, aspects and advantages of the present invention will become more apparent from the following detailed 30 description of the present invention when taken in conjunction with the accompanying drawings.

#### BRIEF DESCRIPTION OF THE DRAWINGS

Fig. 1 is a schematic block diagram illustrating a first configuration

for speeding up a calculation in a power-residue calculating unit.

Fig. 2 is a schematic block diagram illustrating a second configuration for speeding up a calculation in a power-residue calculating unit.

5 Fig. 3 is a schematic block diagram illustrating a configuration of a power-residue calculating unit 1200 in accordance with a first embodiment of the present invention.

Fig. 4 is a schematic block diagram illustrating a configuration of a power-residue calculating unit 1300 in accordance with a second embodiment of the present invention.

10 Fig. 5 is a schematic block diagram illustrating a configuration of a power-residue calculating unit 1400 in accordance with a third embodiment of the present invention.

15 Fig. 6 is a schematic block diagram illustrating a configuration of a power-residue calculating unit 1500 in accordance with a fourth embodiment of the present invention.

Fig. 7 is a schematic block diagram illustrating a configuration of a power-residue calculating unit 1600 in accordance with a fifth embodiment of the present invention.

20 Fig. 8 is a schematic block diagram illustrating a configuration of a power-residue calculating unit 1700 in accordance with a sixth embodiment of the present invention.

25 Fig. 9 is a schematic block diagram illustrating a configuration of a power-residue calculating unit 1800 in accordance with a seventh embodiment of the present invention.

Fig. 10 is a schematic block diagram illustrating a configuration of a power-residue calculating unit 1900 in accordance with an eighth embodiment of the present invention.

#### DESCRIPTION OF THE PREFERRED EMBODIMENTS

30 First Embodiment

(Configuration for Speeding Up Calculation)

A configuration of a power-residue calculating unit of a first embodiment that is less susceptible to Power Analysis will be described

based on first and second configurations for speeding up the calculation in the power-residue calculating unit as described below.

[First Hardware Configuration of Power-Residue Calculating Unit]

Referring to Fig. 1, a power-residue calculating unit 1000 having a first configuration for speeding up the calculation of the power-residue calculating unit includes: an I/F (interface) circuit 101 interfacing with an external bus; an e register 102 holding a key e; a Y register 103 holding a multiplier Y for Montgomery conversion; an N register 104 holding a key N; a B2N register 105 holding a value of 2B+N calculated during Montgomery conversion; an X register 106 holding a plaintext X; a calculating circuit 107 performing calculations for encryption and decryption; a P register 108 holding a calculation result P; and a power-residue control circuit 109 serving as a state machine when a power-residue calculation is executed.

Power-residue calculating circuit 1000 further includes: a Montgomery multiplication residue/residue control circuit 110 serving as a state machine when a Montgomery multiplication residue calculation and a residue calculation are executed; an addition/subtraction control circuit 111 controlling calculations of addition and subtraction; a mode register 112 holding various modes; a command register 113 holding a command; a status register 114 holding a status; an internal bus 115 connected to interface circuit 101, e register 102 and Y register 103 for communicating data between various registers; and an internal bus 116 connected to N register 104, B2N register 105, X register 106, calculating circuit 107, and P register 108 for communicating data between various registers and calculating circuit 107; and a bus dividing circuit 117 connected to internal buses 115 and 116 for controlling data input/output between internal buses 115 and 116.

Power-residue calculating circuit 1000 further includes an Ej/Yj detecting portion 118 between internal bus 115 and power-residue control circuit 109 and Montgomery multiplication residue/residue control circuit 110.

[Calculation Method of Montgomery Calculation]

In performing power-residue calculations, Montgomery-method

multiplication residue calculations and residue calculations are used to attain a high-speed. Among others, a calculation method of the conditional Montgomery calculation [if  $e_j = 1$  then  $Y = MR(X \cdot Y)$ ] in equation (12) will now be described.

5        In executing a loop of equations (12) and (13), key e held in e register 102 is read onto internal bus 115. At the same time, power-residue control circuit 109 instructs  $E_j/Y_j$  detecting portion 118 of bit  $j$  to be read of key e depending on the state of the counter included therein.  $E_j/Y_j$  detecting unit 118 reads  $j$ -th bit value  $e_j$  of key e read out onto internal bus 115 and  
10      provides the same to power-residue control circuit 109.

As multiplier A is right-shifted bit by bit because of equation (21), Amod 2 in equation (20) corresponds to  $j$ -th bit of multiplier A, that is,  $j$ -th bit  $Y_j$  of multiplier Y held in Y register 103. Therefore in performing a loop of equations (19)-(21), multiplier Y held in Y register 103 is read out onto internal bus 115. At the same time, Montgomery multiplication residue/residue control circuit 110 instructs  $E_j/Y_j$  detecting portion 118 of bit  $j$  to be read of multiplier Y depending on the state of the counter included therein.  $E_j/Y_j$  detecting portion 118 reads  $j$ -th bit value  $Y_j$  of multiplier Y read out onto internal bus 115 and provides the same to Montgomery multiplication residue/residue control circuit 110.  
15  
20

The similar process is also performed for residue calculations.

In a multiplication calculation, bus dividing circuit 117 turns off and multiplier Y is sent to  $E_j/Y_j$  detecting portion 118. In a square calculation, multiplicand Y is sent to calculating unit 107 with bus dividing circuit 117 turning on and also to  $E_j/Y_j$  detecting portion 118 as a multiplier at the same time.  
25

#### [Second Hardware Configuration of Power-Residue Calculating Unit]

Power-residue calculating unit 1100 with a modified configuration of power-residue calculating unit 1000 shown in Fig. 1 will now be described.  
30

Fig. 2 is a schematic block diagram illustrating a configuration of power-residue calculating unit 1100 as a second configuration for speeding up the calculation of the power-residue calculating unit.

In the following, the configuration of power-residue calculating unit 1100 in Fig. 2 will be described by contrast with the configuration of power-residue calculating unit 1000 in Fig. 1.

5 In power-residue calculating unit 1100 in Fig. 2, e register 102, Y register 103, X register 106, B2N register 105, N register 104, P register 108, and Ej/Yj detecting portion 118 are similar to the components of power-residue calculating unit 1000 in Fig. 1.

10 A control circuit 100 of power-residue calculating unit 1100 includes three circuits of power-residue calculating unit 1000 in Fig. 1, which are power-residue control circuit 109, Montgomery multiplication residue/residue control circuit 110 and addition/subtraction control circuit 111.

15 Here, a 0 register 130 of power-residue calculating unit 1100 is a register holding a value 0 and is controlled by a signal 0rd from control circuit 100 such that it is selected for the value to be read when X register 106, B2N register 105 and N register 104 are not selected. Note that Y register 103 is selected through bus dividing circuit 117 in place of X register 106 in square calculations. Specifically, in square calculations, selection is made from 0 register 130, Y register 103, B2N register 105, and N register 20 104. In the following description, X register 106 in the process of multiplication should be read as Y register 103 in square calculations.

25 Three circuits of power-residue calculating unit 1100, i.e. a calculating unit 107.1, a carry determining portion 107.2 and a right shift circuit 107.3 correspond to calculating circuit 107 in power-residue calculating unit 1000 in Fig. 1.

A register 140 is a register holding a value 0 and is controlled by signal 0rd from control circuit 100 such that it is selected for data to be read when P register 108 is not selected.

30 A Wr inhibit processing unit 122 inhibits a write into Y register 103 by masking a write signal Ywr to Y register 103 during execution of the if statement described later where (ej) = 0 in the conditional expression.

A correction calculation signal from control circuit 100 to N register 104 is provided to N register 104 through an OR gate 128 for instructing

that the value in N register 104 is read when a correction calculation is executed.

Wr inhibit processing portion 134 inhibits a write into a P register 108 by masking a write signal Pwr to P register 108 when no correction is needed in execution of the correction calculation described later.

Note that a signal Xxwr from control circuit 100 is a write signal for an Xx register (E register, Y register, X register, or the like), and a signal Xxrd is a read signal for Xx register.

In Fig. 2, the external bus, interface 101, mode register 112, command register 113, status register 114, and the like are not shown.

The basic operation of power-residue calculating unit 1100 is similar to the operation of power-residue calculating unit 1000. In the following, this operation will be described.

The operation of power-residue calculating unit 1100 shown in Fig. 2 uses a left binary method for a power calculation and the Montgomery method for a multiplication residue calculation.

A result of unit element 1 of multiplication subjected to a Montgomery conversion is set in Y register 103 externally, for example, from a CPU (Central Processing Unit) (step S100). Next, a result of an input text subjected to a Montgomery conversion is set in register X 106 externally (for example from CPU) (step S102).

A modulus is set in N register 104 externally (for example from CPU) (step S104).

When R is a constant  $2^{1024}$  for Montgomery conversion, the following two lines are repeated 1024 times ( $j = 1023$  to 0) (step S106).

$Y = Y \times Y/R \bmod N$  ... square calculation  
If  $e_j = 1$  then  $Y = X \times Y/R \bmod N$  else void  $X \times Y/R \bmod N$   
...multiplication

Thereafter the value of Y undergoes a Montgomery inverse conversion ( $Y = 1 \times Y/R \bmod N$ ) (step S108).

The result of the Montgomery inverse conversion in Y register 103

will be a value of a result of power-residue calculation (step S110).

Here, depending on whether the aforementioned if statement is true or false, write signal Ywr for writing the multiplication result described above into Y register 103 is masked.

5 In the operation of power-residue calculating unit 1100, the following correction calculation is performed in the end of each Montgomery multiplication residue processing such that the result Y of the Montgomery multiplication residue processing falls within the range of  $N > Y \geq 0$ :

if  $P < 0$  then  $P = P + N$ .

10 As a result, unlike the operation of power-residue calculating unit 1000, the residue calculation (equation (15)) for putting the result within this range is not executed.

(Montgomery Multiplication Residue Calculation Process of Power-Residue Calculating Unit 1100)

15 The Montgomery multiplication residue calculation process of power-residue calculating unit 1100 will now be described briefly, where  $Y = X \times Y / R \bmod N$  and where  $B = N - X$  is executed in the beginning of the Montgomery multiplication residue processing.

20 A modulus is read from N register 104 with signal Nrd and written into P register 108 with signal Pwr (step S200).

Then X register 106 is read with signal Xrd and P register 108 is read with signal Prd. Calculating unit 107.1 executes a subtraction  $(P - N)$  to write the subtraction result into P register 108 with signal Pwr without right-shift (step S202).

25 P register 108 is then read with signal Prd so that the subtraction result is written into B2N register 105 with a signal Bwr (not shown) from control circuit 100 (step S204). An initial value in P register 108 is assumed 0 (step S206).

30 Thereafter the following calculation is repeated 1024 times ( $j = 0$  to 1023) (step S208).

i) The value in X register 106, the value in N register 104, the value in B2N register 105, and the value in 0 register 130, selected in accordance with  $Y_j$  and  $P_0, X_0$ , are read in response to signal Xrd, Nrd, Brd, or Ord (step

S210).

ii) An addition/subtraction is executed in calculating unit 107.1 and the result is right-shifted by one bit in right-shift circuit 107.3 and is then written into P register 108 with signal Pwr (step S212).

5 Thereafter the following correction calculation is executed (step S214).

The value in N register 104 is read. The calculation  $(P + N)$  is executed in calculating unit 107.1. Here a right-shift is not performed. If the previous value in P register 108 is negative, the result of the calculation  $(P + N)$  is written into P register 108 with signal Pwr. On the other hand if the previous value in P register 108 is 0 or positive, signal Pwr is masked so that the result of the calculation  $(P + N)$  is not written into P register 108.

In the foregoing, the description has been made briefly. The operation of power-residue calculating unit 1100 will be described more specifically in the following.

The power-residue calculation ( $Y = X^e \bmod N$ ) has the following process using the left binary method, for example, where a key is of 1024 bits.

```

20      Y = 1           // set initial value
        for j = 1023, 0
            Y = Y×Y mod N    // square residue
            If e[j]  then  Y = X×Y mod N    // multiplication residue
        End for
    
```

25 Here,  $e[j]$  is a  $j$ -th bit value of  $e$ . The calculation result is assumed as the value in X register 103.

The following two calculations can be selectively performed without changing a large part of hardware only by selecting X or Y as a multiplicand in the multiplication residue processing for use in the left binary method.

$$Y = Y \times Y \bmod N$$

In the right binary method,  $X = Y^e \bmod N$  can be obtained by the following process.

5             $X = 1$             // set initial value  
for  $j = 0, 1023$   
if  $e[j]$  then  $X = X \times Y \bmod N$             // multiplication residue  
 $Y = Y \times Y \bmod N$             // square residue  
End for

10            Here the calculation result is assumed as the value in X register 106.  
The following two calculations can be selectively performed without  
changing a large part of hardware only by adding a function of selecting a  
register to store the result to a function of selecting a multiplicand in the  
15            multiplication residue processing for use in the right binary method.

$Y = Y \times Y \bmod N$   
 $X = X \times Y \bmod N$

20            Meanwhile, the aforementioned Montgomery method ( $Y = Y \cdot X/R \bmod N$ ) has the following process where Y has a 1024-bit length again.

25             $P = 0$             // set initial value  
For  $j = 0$  to 1023  
 $M = (P + Y[j] \cdot X) \bmod 2 = P[0]^Y[j] \cdot X[0]$   
 $P = (P + Y[j] \cdot X + M \cdot N)/2$             // accumulative addition  
End for  
 $P = P - N$             ... (22) // subtraction  
If  $P < 0$  then  $P = P + N$             // correction calculation  
30             $Y = P$             // store the result

In power-residue calculating unit 1000 shown in Fig. 1, the hardware performing the Montgomery method is implemented by

calculating the second and third terms in the accumulative addition beforehand and changing a two-term addition to a three-term addition, thereby reducing the hardware for the adder. In other words,  $B = 2X + N$  is calculated.

5 On the other hand, in the configuration implementing the process including equation (22) described above,

$$B = X + N$$

is calculated beforehand and stored in a register, thereby reducing the hardware for the adder.

10 In other words, since  $M = P[0]^Y[j] \cdot X[0]$ , a register is selected in accordance with  $M$  and  $Y[j]$  ( $j$ -th bit value of  $Y$ ) as follows for execution of the accumulative addition.

15 It is noted that in the following the value stored in  $P$  register 108 is represented by "P", the value stored in  $X$  register 106 is represented by "X", the value stored in  $N$  register 104 is represented by "N", and the value stored in  $B2N$  register 105 is represented by "B".

	$Y[j]$	$P[0]^Y[j] \cdot X[0]$	rd signal	accumulative addition
20	0	0	0rd	$(P+0)/2$
	1	0	Xrd	$(P+X)/2$
	0	1	Nrd	$(P+N)/2$
	1	1	Brd	$(P+B)/2$

25 The Montgomery method ( $Y = Y \cdot X/R \bmod N$ ) described above can be configured as follows to eliminate  $P = P - N$  in equation (22). The brief description after (Montgomery Multiplication Residue Calculation Process of Power-Residue Calculating Unit 1100) illustrates an example of this process.

30  $P = 0$  // set initial value  
For  $j = 0$  to 1023  
 $M = (P+Y[j] \cdot X) \bmod 2 = P[0]^Y[j] \cdot X[0]$   
 $P = (P+Y[j] \cdot X - M \cdot N) / 2$  // accumulative addition/subtraction

End for

If  $P < 0$  then  $P = P+N$  // correction calculation  
 $Y = P$  // store the result

5 In this case, the following calculation may be performed in the addition/subtraction by storing in a register  $B = N - X$  calculated beforehand for the second and third terms of the addition/subtraction.

$$P = (P - (N - X))/2$$

10

Alternatively, the following calculation may be performed in the addition/subtraction by storing  $B = X - N$  in a register beforehand.

$$P = (P + (X - N))/2$$

15

Either of the methods described above can reduce the hardware amount for the adder. Specifically, as  $M = P[0] \wedge Y[j] \cdot X[0]$ , the accumulative addition/subtraction may be executed by selecting a register in accordance with  $M$  and  $Y[j]$  and performing addition or subtraction as follows.

20

	$Y[j]$	$M$	$+/-$	rd signal	accumulative addition
	0	0	+	0rd	$(P+0)/2$
	1	0	+	Xrd	$(P+X)/2$
	0	1	-	Nrd	$(P-N)/2$
25	1	1	-	Brd	$(P-B)/2$

Here,  $B = N - X$ .

The Montgomery method ( $Y = Y \cdot X/R \bmod N$ ) may be executed by replacing  $X$  with  $Y$ . In other words, Xrd signal to X register 106 is directed to Y register 103 and is read through bus dividing circuit 117 for execution.

Furthermore, "/2" in the accumulative addition/subtraction is performed by a right shift process. The correction calculation can also be performed as follows with a constant execution time irrespective of input

data by masking the write signal in a manner similar to the binary method.

if  $P < 0$  then  $P = P+N$  else void  $P+N$

5 More specifically, in a correction calculation, Nrd signal may be provided to N register 104 and write signal Pwr to P register 108 may be masked in accordance with a signal indicative of whether a correction is required or not.

10 It is noted that in Fig. 2 the selection of Xxrd signal in accordance with M, Yj, the setting of initial value  $P = 0$ , the calculation and setting of the value to the register storing the value of B, and the like are processed by producing an appropriate control signal in control circuit 100 and providing the signal to the calculating unit. In Fig. 2, the initial value for the power-residue processing is provided, for example, from CPU.

15 [Configuration of Power-Residue Calculating Unit 1200 of First Embodiment]

20 Under the preparation described above, a power-residue calculating unit 1200 configured in a manner even less susceptible to Power Analysis will be described below based on the configuration of power-residue calculating unit 1100.

Fig. 3 is a schematic block diagram illustrating the configuration of power-residue calculating unit 1200 in accordance with the first embodiment.

25 In power-residue calculating unit 1200, as described below, a K register 132 in Fig. 2 is added to power-residue calculating unit 1100 in Fig. 2 to perform a process of

if  $e_j = 1$   $Y = X \times Y/R \bmod N$  else  $K = X \times Y/R \bmod N$   
in place of the process of

if  $e_j = 1$   $Y = X \times Y/R \bmod N$  else void  $X \times Y/R \bmod N$   
30 in the power-residue calculation.

Fig. 3 shows a configuration in the case of the left binary method by way of example.

Referring to Fig. 3, the operation of the additional K register 132 will

be detailed.

In the binary method of the power-residue calculation, K register 132 is used as a register for storing a result when the determination of the if statement is "false" during the execution of the if statement described above.

5 Note that the value of  $e[j]$  is verified during the square calculation of Y and defined before the if statement is processed.

Here the process of  $Y = X^e \bmod N$  is performed for 1024-bit data as follows.

10  $Y = 1$   
For  $j = 1023$  to 0  
     $Y = Y \cdot Y \bmod N$   
    If ( $e[j]=1$ ) then  $Y = Y \cdot X \bmod N$  else  $K = Y \cdot X \bmod N$   
End for

15 The Montgomery method is used in the above process as follows.

20  $X = X \cdot R \bmod N$   
 $Y = R \bmod N$   
For  $j = 1023$  to 0  
     $Y = Y \cdot Y/R \bmod N$   
    If ( $e[j]=1$ ) then  $Y = Y \cdot X/R \bmod N$  else  $K = Y \cdot X/R \bmod N$   
End for  
 $Y = Y/R \bmod N$

25 Here signal  $e[j]$  from  $Ej/Yj$  detecting portion 118 is used to mask write signal  $Ywr$  to Y register 103.

When the if statement is executed, AND between an inverted signal of a signal (an inverted signal of signal  $e[j]$ ) output from Wr inhibit processing portion 122 in response to signal  $e[j]$  and write signal  $Ywr$  from control circuit 100 to Y register 103 is calculated in gate circuit 124, result of which is provided to Y register 103 as write signal  $Ywr'$ .

AND between a signal (an inverted signal of signal  $e[j]$ ) output from

Wr inhibit processing portion 122 in response to signal  $e[j]$  and write signal  $Y_{wr}$  to Y register 103 is calculated in gate circuit 126, result of which is provided to K register 132 as write signal  $K_{wr}$ .

5 It is noted that the similar process can be performed with the right binary method.

10 By configuring power-residue calculating unit 1200 as described above, a write operation for the register is performed similarly whether the condition of the if statement is "true" or "false". Therefore in any case current in a write operation flows so that the immunity against Power Analysis is improved.

[Second Embodiment]

Fig. 4 is a schematic block diagram illustrating a configuration of power-residue calculating unit 1300 in accordance with a second embodiment of the present invention.

15 As described below, B2N register 105 is used as a register for storing intermediate data B to perform a process of

if  $e_j = 1$   $Y = X \times Y/R \bmod N$  else  $B = X \times Y/R \bmod N$   
in place of the process of

if  $e_j = 1$   $Y = X \times Y/R \bmod N$  else void  $X \times Y/R \bmod N$   
20 during the power-residue calculation.

In other words, in the if statement with logic "false", register 105 also serves as a register performing a dummy process (storing intermediate data B), rather than providing a separate register for storing data as a dummy process.

25 It is noted that in Fig. 4 a correction determining portion 150 outputting a signal based on whether a correction calculation is required or not and a gate circuit 138' are used in place of Wr inhibit processing portion 134 and gate circuit 138 in Fig. 3.

30 B2N register 105 is a temporary register for use when the Montgomery multiplication residue is calculated, and may store data without any problems when the logic of the if statement is "false".

The process of  $Y = X^e \bmod N$  is performed for data of 1024 bits as follows. It is noted that the value of data in B2N register 105 is assumed as

B2N.

5            $Y = 1$   
For  $j = 1023$  to 0  
     $Y = Y \cdot Y \bmod N$   
    If ( $e[j]=1$ ) then  $Y = Y \cdot X \bmod N$  else  $B2N = Y \cdot X \bmod N$   
End for

10           The Montgomery method is used in the above process as follows.

15            $X = X \cdot R \bmod N$   
           $Y = R \bmod N$   
For  $j = 1023$  to 0  
     $Y = Y \cdot Y/R \bmod N$   
    If ( $e[j]=1$ ) then  $Y = Y \cdot X/R \bmod N$  else  $B2N = Y \cdot X/R \bmod N$   
End for  
     $Y = Y/R \bmod N$

20           Here signal  $e[j]$  from  $Ej/Yj$  detecting portion 118 is used to mask  
write signal  $Y_{wr}$  to  $Y$  register 103.

25           When the if statement is executed, AND between an inverted signal  
of a signal (an inverted signal of signal  $e[j]$ ) output from  $Wr$  inhibit  
processing portion 122 in response to signal  $e[j]$  and write signal  $Y_{wr}$  from  
control circuit to  $Y$  register 103 is calculated in gate circuit 124, result of  
which is provided to  $Y$  register 103 as write signal  $Y_{wr}'$ .

30           AND between a signal (an inverted signal of signal  $e[j]$ ) output from  
 $Wr$  inhibit processing portion 122 in response to signal  $e[j]$  and write signal  
 $Y_{wr}$  to  $Y$  register 103 is calculated in gate circuit 126, result of which is  
provided to  $B2N$  register 105 as a write signal  $B2N_{wr}$ .

It is noted that the similar process can be performed with the right  
binary method.

By configuring power-residue calculating unit 1200 as described  
above, a write operation for the register is performed similarly whether the

condition of the if statement is "true" or "false. Therefore in any case current in a write operation flows so that immunity against Power Analysis is improved.

[Third Embodiment]

5 Fig. 5 is a schematic block diagram illustrating a configuration of power-residue calculating unit 1400 in accordance with a third embodiment of the present invention.

10 In the following, the operation of power-residue calculating unit 1400 shown in Fig. 5 will be described by contrast with the operation of power-residue calculating unit 1100 shown in Fig. 2.

First, in the operation of power-residue calculating unit 1400 shown in Fig. 5, the process of

15 "A result of unit element 1 of multiplication subjected to a Montgomery conversion is set in Y register 103 externally, for example, from a CPU (Central Processing Unit) (step S100)"

is omitted from the process in the power-residue calculation in the operation of power-residue calculating unit 1100 shown in Fig. 2.

20 Furthermore, in the operation of the correction calculation in the operation of power-residue calculating unit 1100 shown in Fig. 2, the process in step S214:

25 "The value in N register 104 is read. The calculation  $(P + N)$  is executed in calculating unit 107.1. Here a right-shift is not performed. If the previous value in P register 108 is negative, the result of the calculation  $(P + N)$  is written into P register 108 with signal Pwr. On the other hand if the previous value in P register 108 is 0 or positive, signal Pwr is masked so that the result of the calculation  $(P + N)$  is not written into P register 108."

includes the following additional process:

30 "When  $e[j] = 1$  for the first time, data is read from X register 106 and 0 register 140 in place of N register 104 and P register 108. The calculation  $(0+X)$  is executed in calculating unit 107. Whether the previous value in P register 108 is positive or negative, write signal Pwr allows the result of calculation  $(0+X)$  to be written in P register."

In other words, power-residue calculating unit 1400 of the third

embodiment aims to avoid the following problem.

Namely, where the right binary method is used in the power-residue calculation, the initial value of Y is 1 and the value in Y register 103 does not change until  $e[j] = 1$ . Once  $e[j]=1$  and  $Y = XY$ , the value in Y register 103 becomes dependent on input text X. Therefore the number of consecutive bits 0 from the upper bit of  $e[j]$  can be determined externally by varying input texts and observing the timing at which the current depends on the input text.

Therefore the following process is generally performed in power-residue calculating unit 1400 of the third embodiment.

When the condition that  $e[j] = 1$  for the first time is satisfied,  $YX$  results in X. In other words, if the process for  $Y = X$  is performed at the time of  $e[j] = 1$  for the first time, irrespective of the prior values of Y, ultimately the same result can be obtained. With the use of such a technique, the values of Y prior to the first-time  $e[j] = 1$  become irrelevant so that the initial value of Y can be set to the value dependent on the input text. Therefore with the current dependent on the input text throughout the period of bits of Y, the number of 0s can be unknown.

As shown in Fig. 5, power-residue calculating unit 1400 of the third embodiment is provided with a first-Ej-value detecting portion 161 for detecting the first-time  $e[j]=1$  along with a multiplication circuit capable of altering the multiplication result to X itself in accordance with the detection result.

Fig. 5 shows an example where a Montgomery multiplication residue circuit is used as a multiplication circuit.

The difference between power-residue calculating unit 1000 shown in Fig. 1 and power-residue calculating unit 1400 is summarized as follows.

In power-residue calculating unit 1000 a correction calculation is performed with a residue being obtained collectively in the last place whereas in power-residue calculating unit 1400 a correction calculation is performed at the end of each Montgomery multiplication residue calculation process. Furthermore,  $B = N - X$  is calculated at the beginning of each Montgomery multiplication residue calculation process, and a subtraction is

executed in calculating unit 107 when B register 105 is selected as the input for calculating unit 107 or when N register 104 is selected. Therefore as the result to be corrected is within the range from  $(-N)$  to  $(N - 1)$ , whether the correction calculation is required or not can be determined in carry determining portion 107.2 in calculating unit 107.

5 The processing in power-residue calculating unit 1400 of the third embodiment will be further detailed.

First of all, the operation of first-Ej-value detecting portion 161 is mainly described.

10 In the power-residue processing in power-residue calculating unit 1400, an S register 160 is used as follows. S register 160 has its initial value of 0 and stores 1 when  $EJ (= e[j])$  is 1 each time the loop ends.

S = 0

15 For j = 1023 to 0

Y = Y·Y mod N, EJ = e[j]

If (EJ = 1) then if (S = 1) then Y = Y·X mode N else Y = X  
if EJ = 1 then S = 1

End for

20

The Montgomery method is used in the above process as follows.

X = X·R mod N

S = 0

25 For j = 1023 to 0

Y = Y·Y/R mod N

If (EJ = 1) then if (S = 1) then Y = Y·X/R mod N else Y = X  
if EJ = 1 then S = 1

End for

30 Y = Y/R mod N

As in the process described above, in the power-residue calculating unit 1400 of the third embodiment, the detection signal (L corresponding to

EJ) for the first-time  $e[j] = 1$  allows the multiplication result to be set to X.

In the correction calculation at the end of the Montgomery calculation, detection signal L for the first-time  $e[j] = 1$  allows a value 0 to be selected from 0 register 130 in place of the value in P register 108 or a value in X register 106 to be selected in place of the value in N register 104 or 0 register 130 as an input for calculating unit 107 as follows, thereby resulting in  $P = X$ . Thereafter data transfer is performed for storing the value in P register 108 into Y register 103.

10 // correction calculation

carry	L	Calculation
1	0	$P = P+N$
0	0	$P = P+0$
X	1	$P = 0+X$

15

Thereafter through the process for  $Y = P$  the multiplication residue result that is positively corrected or the value of X is stored in Y register 103.

Here since the initial value of Y stored in Y register 103 does not affect the final result, a variety of setting methods can be employed in consideration of DPA. For example the value left in Y register 103 in the prior process can be used as it is.

Alternatively, for example, the result after reduction is written in Y register 103 and the result after the Montgomery conversion is written in X register 106, followed by execution of the binary method. Then  $Y = Y \times Y$  will be  $(M \bmod N) \times (MR \bmod N) / R \bmod N$ , and thus  $M \times M \bmod N$  where R is missing, resulting in a different input. Therefore the matching pattern in the binary method rarely appears.

In the configuration above, S register 160 holding data indicative of whether the first 1 appears is provided. Therefore it becomes unnecessary to set a unit element at the beginning by setting an initial value during the correction calculation for the Montgomery calculation when the first 1 appears. As a result, the number of consecutive 0s at the beginning is hardly discovered as the pattern of current consumption indicative of

whether the dependency of input data exists. Such a process can be realized only by switching the operand of the Montgomery correction calculation, with less effect on the current and with the same execution cycle.

5 [Fourth Embodiment]

Fig. 6 is a schematic block diagram illustrating a configuration of a power-residue calculating unit 1500 in accordance with a fourth embodiment of the present invention.

10 In the following, the operation of power-residue calculating unit 1500 shown in Fig. 6 will be described by contrast with the operation of power-residue calculating unit 1100 shown in Fig. 2.

15 In the operation of power-residue calculating unit 1500 shown in Fig. 6, the following process in the Montgomery multiplication residue calculation in the operation of power-residue calculating unit 1100 shown in Fig. 2 will be omitted:

"An initial value of P register 108 is set to 0 (step S206)."

Furthermore, in the following process in the Montgomery multiplication residue calculation of power-residue calculating unit 1100 shown in Fig. 2:

20 "Thereafter the following calculation is repeated 1024 times ( $j = 0$  to 1023) (step S208).

i) The value in X register 106, the value in N register 104, the value in B2N register 105, and the value in 0 register 130, selected in accordance with  $Y_j$  and  $P_0$ ,  $X_0$ , are read in response to signal  $X_{rd}$ ,  $N_{rd}$ ,  $B_{rd}$ , or  $0_{rd}$  (step 25 S210).

ii) An addition/subtraction is executed in calculating unit 107.1 and the result is right-shifted by one bit in right-shift circuit 107.3 and is then written into P register 108 with signal Pwr (step S212)",

30 the value read from P register 108 is subjected to addition/subtraction. On the other hand, in power-residue calculating unit 1500 shown in Fig. 6, the process is changed as follows:

"when  $Y_j = 1$  for the first time in 1024 times, the value in 0 register 140 is read instead of P register 108."

In other words, power-residue calculating unit 1500 of the fourth embodiment aims to avoid the following problem.

Namely, in the case of a Montgomery multiplication residue calculation or a multiplication, X is added when  $Y[j] = 1$  with data in P register 108 being right-shifted. In doing so, the initial value of P is set to 0. Therefore the value in P register 108 is not changed until  $Y[j] = 1$ . Once  $Y[j] = 1$ , the value to be written in P register 108 varies for each iteration as the iteration includes the right-shift. Therefore the number of consecutive 0 bits from the lower bit of  $Y[j]$  can be determined for each input text by 5 changing input texts and observing the timing at which the current depends on the input text.

10

Then power-residue calculating unit 1500 of the fourth embodiment generally performs the following process.

The value in P register is 0 for the first-time  $Y[j] = 1$ . Specifically, 15 irrespective of the prior values of P, if 0 is read from P register 108 at that time, ultimately the same result can be obtained. With the use of such a technique, the value in P register 108 prior to  $Y[j] = 1$  will be irrelevant to the final result. Therefore the initial value in P register 108 can be set to a value dependent on the input text, and with the current dependent on the 20 input text throughout the period of bits of  $Y[j]$ , the number of 0s can be externally unknown by Power Analysis.

As shown in Fig. 6, power-residue calculating unit 1500 of the fourth embodiment is provided with a first- $Y[j]$  detecting portion 180 for detecting  $Y[j] = 1$  for the first time, and a configuration allowing 0 to be read from 0 25 register 140 in place of data to be read from P register 108 in accordance with the detection result.

Power-residue calculating unit 1500 of the fourth embodiment in Fig. 6 is a Montgomery multiplication residue circuit.

As shown in Fig. 6, in power-residue calculating unit 1500, the 30 output of first- $Y[j]$  detecting portion 180 is assumed as H.

In the power-residue process, a T register 182 is used in a following manner. T register 182 has its initial value of 0 and stores 1 when  $Y[j] = 1$  each time a process loop ends. Output H is "1" when (T=0) and ( $Y[j] = 1$ ).

```

T = 0
For j = 0 to 1023
    If H = 0  then
5      M = (P+Y[j]·X) mod 2
        P = (P+Y[j]·X+M·N)/2
    Else
        M = (0+Y[j]·X) mod 2
        P = (0+Y[j]·X+M·N) / 2
10   If Y[j] = 1  then  T = 1
    End for

```

As shown in Fig. 6, during the Montgomery calculation, the detection signal H indicating that the first-time  $Y[j] = 1$  is detected is applied to one inputs of gate circuits 172 and 174. As a result, the value in 0 register 140 is selected in place of the value in P register 108 as an input of calculating unit 107 as follows, resulting in  $P = X$ .

20	// during Montgomery calculation	
	H	content of calculation
	0	$M = (P+Y[j]·X) \text{ mod } 2$
	0	$P = (P+Y[j]·X+M·N)/2$
	1	$M = (0+Y[j]·X) \text{ mod } 2$
	1	$P = (0+Y[j]·X+M·N)/2$

Furthermore, as an example of the initial value for P register 108, the previous calculation result can be used as it is without initializing P register 108 in the Montgomery multiplication residue processing.

Because of the configuration as described above, when  $Y[j]$  is calculated from LSB (Least Significant Bit) in the multiplication and the Montgomery multiplication residue calculation, less dependency from the consumed current can be prevented until the first 1 appears. Therefore in the value P to be stored in P register 108, the number of consecutive 0s from

LSB can hardly be discovered from the current variation pattern.

[Fifth Embodiment]

Fig. 7 is a schematic block diagram illustrating a configuration of power-residue calculating unit 1600 in accordance with a fifth embodiment of the present invention.

In the following, the operation of power-residue calculating unit 1600 shown in Fig. 7 will be described by contrast with the operation of power-residue calculating unit 1100 shown in Fig. 2.

First, in the operation of power-residue calculating unit 1600 shown in Fig. 7, the following step S214 during the correction calculation in the operation of power-residue calculating unit 1100 shown in Fig. 2:

"Thereafter the following correction calculation is executed (step S214).

The value in N register 104 is read. The calculation  $(P + N)$  is executed in calculating unit 107.1. Here a right-shift is not performed. If the previous value in P register 108 is negative, the result of the calculation  $(P + N)$  is written into P register 108 with signal Pwr. On the other hand if the previous value in P register 108 is 0 or positive, signal Pwr is masked so that the result of the calculation  $(P + N)$  is not written into P register 108."

is changed as follows:

"Thereafter the following correction calculation is executed (step S214').

If the previous P is negative, data is read from N register 104, and if the previous P is 0 or positive, 0 is read from O register 130. The calculation result is written into P register 108 without write signal Pwr being masked."

In other words, power-residue calculating unit 1600 of the fifth embodiment aims to avoid the following problem.

Generally, calculating unit 107, P register 108 and the like are shared in order to realize the correction calculation with less additional hardware. In this case, since the value before the correction is in P register 108, the correction calculation may be executed for writing into P register 108 only when the correction is required. Alternatively, in order to avoid the time variations resulting from the presence/absence of the correction

calculation, the correction calculation may be executed whether the correction is required or not, and the write signal to P register may be masked when the correction is not required. With such a configuration, however, the consumed current varies depending on the presence/absence of the write signal to P register.

Power-residue calculating unit 1600 shown in Fig. 7 has a modified configuration of power-residue calculating unit 1300 of the second embodiment. Signals Nrd and Ord controlling the reads of N register 104 and 0 register 130 are applied to one inputs of OR gates 184 and 186, respectively. The other inputs of OR gates 184 and 186 receive the outputs from correction determining portion 150 that determines whether the correction is required or not. Power-residue calculating unit 1600 does not include gate circuit 138' controlling the write into P register 108 depending on whether the correction is required or not.

More specifically, during the correction calculation, based on value P in P register 108 being positive/negative, that is, the carry of calculating unit 107.1 performing addition/subtraction, N register is selected and the calculation ( $P + N$ ) is executed if the carry is negative. On the other hand, if the carry is positive, 0 register 130 is selected and the calculation ( $P + 0$ ) is executed. In either case, the calculation result is written into P register 108 so that the variations in the consumed current due to the presence/absence of the write can be avoided.

Data read from P register 108 is then stored in Y register 103.

With such a configuration, even if the correction is not required, the result is always written into the register. As a result, the variations in the consumed current resulting from the presence and absence of the write into the register can be avoided. Therefore it is less likely that the processing is externally estimated using Power Analysis.

#### [Sixth Embodiment]

Fig. 8 is a schematic block diagram illustrating a configuration of power-residue calculating unit 1700 in accordance with a sixth embodiment of the present invention.

In the following, the operation of power-residue calculating unit

1700 shown in Fig. 8 will be described by contrast with the operation of power-residue calculating unit 1600 shown in Fig. 7.

In the operation of power-residue calculating unit 1700 shown in Fig. 8, the following step S214' during the correction calculation in the operation of power-residue calculating unit 1600 shown in Fig. 7:

5 "Thereafter the following correction calculation is executed (step S214').

If the previous P is negative, data is read from N register 104, and if the previous P is 0 or positive, 0 is read from 0 register 130. The calculation result is written into P register 108 without write signal Pwr being masked."

10 is changed as follows:

"Thereafter the following correction calculation is executed (step S214").

15 If the previous P is negative, data is read from N register 104, and if the previous P is 0 or positive, 0 is read from 0 register 130. The calculation result is right-shifted and written into P register 108 without write signal Pwr being masked.

When the result is read from P register 108, it is restored by left-shift."

20 In other words, power-residue calculating unit 1700 of the sixth embodiment aims to avoid the following problem.

Namely, in power-residue calculating unit 1600 of the fifth embodiment, the current may still have slight variations because the same value ( $P + 0$ ) as stored in P or the different value ( $P + N$ ) is written into P register 108. Therefore there may be a possibility of Power Analysis.

25 In order to eliminate such a problem, in power-residue calculating unit 1700 of the sixth embodiment, the right shift process is performed after the addition/subtraction in the correction calculation, though the right shift processing is not required, so that the value to be written into P register 108 after completion of the correction calculation may be varied even if the correction is not required. Therefore the consumed current has even less variations whether the correction is required or not.

The overflowing LSB during the right shift operations are stored in a

V register 190 that is separately provided. When the value in P register 108 is transferred to another register, the correct value is restored by executing the left shift process and inputting the LSB value stored in V register 190.

5 Power-residue calculating unit 1700 shown in Fig. 8 is provided, in addition to the configuration of power-residue calculating unit 1600 of the fifth embodiment, with V register 190 that stores the carry in the right shift process performed after the process in calculating unit 107, and providing the stored value for the left shift process performed on the data that has  
10 been read from P register 108. Right shift circuit 107.3 is controlled by an OR circuit 188 outputting a signal to be active when control circuit 100 provides the instructions for the correction calculation and the right shift process. Power-residue calculating unit 1700 is further provided with a left shift circuit 192 for performing the left shift process described above.

15 P register 108 stores a value different from the value originally stored in P register 108 (the right-shifted value) by executing the correction calculation, writing LSB into V register 190 and performing the right shift process.

20 When data is read from P register 108 after the execution of the correction calculation, a left shift is performed in left shift circuit 192 and the value in V register 190 is input for LSB. That value is stored in Y register 103.

25 With the configuration as described above, the operation of writing the same value as originally stored in P register 108 is written into P register 108 again in the correction calculation can be eliminated, and the increased/decreased current depending on the presence/absence of the correction can be avoided in the process of writing into P register 108.

#### [Seventh Embodiment]

30 Fig. 9 is a schematic block diagram illustrating a configuration of a power-residue calculating unit 1800 in accordance with a seventh embodiment of the present invention.

In the following, the operation of power-residue calculating unit 1800 shown in Fig. 9 will be described by contrast with the operation of

power-residue calculating unit 1100 shown in Fig. 2.

First, in the operation of power-residue calculating unit 1800 shown in Fig. 9, the following step S214 during the correction calculation in the operation of power-residue calculating unit 1100 shown in Fig. 2:

5 "Thereafter the following correction calculation is executed (step S214).

The value in N register 104 is read. The calculation  $(P + N)$  is executed in calculating unit 107.1. Here a right-shift is not performed. If the previous value in P register 108 is negative, the result of the calculation 10  $(P + N)$  is written into P register 108 with signal Pwr. On the other hand if the previous value in P register 108 is 0 or positive, signal Pwr is masked so that the result of the calculation  $(P + N)$  is not written into P register 108."

is changed as follows:

15 "Thereafter the following correction calculation is executed (step S214").

The value in N register 104 is read. The calculation  $(P + N)$  is executed in calculating unit 107.1. Here a right shift is not performed. If the previous value in P register 108 is negative, the result of the calculation 20  $(P + N)$  is written into P register 108 with signal Pwr. On the other hand, if the previous value in P register 108 is 0 or positive, the result of the calculation  $(P + N)$  is written into a separate Q register 194."

Therefore power-residue calculating unit 1800 of the seventh embodiment also aims to avoid the same problem as power-residue calculating unit 1700 of the sixth embodiment aims to avoid.

25 In order to eliminate such a problem, power-residue calculating unit 1800 of the seventh embodiment 1800 is provided, in addition to the configuration of power-residue calculating unit 1200 of the first embodiment, with Q register 194 and a gate circuit 196 receiving the output from Wr inhibit processing portion 134 and write signal Pwr to be ANDed for 30 producing a signal controlling a write into Q register 194.

In such a configuration, the correction calculation  $(P + N)$  itself is always executed whether the correction calculation is required or not, and Q register 194 is provided for storing the result even if the correction

calculation is not required. The calculation result ( $P + N$ ) is written into Q register 194 depending on whether the correction is required or not in a manner mutually exclusive to P register 108.

5 In the configuration as described above, the value of the calculation value ( $P + N$ ) is normally different from both the value originally stored in Q register 194 and the value originally stored in P register 108. This can reduce the current difference depending on whether the value to be written into P register 108 is same as the value originally stored in P register 108

#### [Eighth Embodiment]

10 Fig. 10 is a schematic block diagram illustrating a configuration of power-residue calculating unit 1900 in accordance with an eighth embodiment of the present invention.

15 The aim of power-residue calculating unit 1900 of the eighth embodiment is as follows. In the case of the left binary, the value of multiplier Y is different between the multiplication and the following square calculation for  $e[j] = 1$ , while that value is same for  $e[j] = 0$ . In the case of the right binary, the value of multiplicand X is different between the multiplication and the following multiplication for  $e[j] = 0$ , while that value is same for  $e[j] = 1$ . Therefore the same pattern is included in the consumed 20 current only when  $e[j] = 0$ , resulting in a possible susceptibility to Power Analysis. Power-residue calculating unit 1900 of the eighth embodiment eliminates this possibility.

25 In power-residue calculating unit 1900 of the eighth embodiment shown in Fig. 10, the configuration of power-residue calculating unit 1200 of the first embodiment is modified such that it is adapted to the right binary method described above and the multiplicand is set to the value in K register 132 rather than to X when the determination of the if statement is "false".

30 Therefore, as compared to power-residue calculating unit 1200, in power-residue calculating unit 1900, gate circuit 124 is omitted and signal Ywr is applied to Y register 103 as it is. In addition, provided are a gate circuit 210 providing to X register 106 AND between inverted level of the signal from Wr inhibit processing portion 122 and signal Xwr, and a gate circuit 212 providing to X register 106 AND between an inverted level of the

signal from Wr inhibit processing portion 122 and signal Xrd.

Gate circuit 126 receives signal Xrd in place of signal Ywr. Multiplication residue calculating circuit 1900 is further provided with a gate circuit 214 providing to K register 132 AND between the signal from Wr 5 inhibit processing portion 122 and signal Xrd.

The remaining configuration is similar to the configuration of multiplication residue calculating circuit 1200 of the first embodiment. Therefore the same components will be denoted with the same reference characters and the description thereof will not be repeated.

10 In the process of this power-residue calculating unit 1900, the iterative portion in the binary method is as follows:

for  $j = 0$  to 1023

15 if  $e[j] = 1$  then  $X = X \times Y/R \bmod N$  else  $K = K \times Y/R \bmod N$

$Y = Y \times Y/R \bmod N$

end for.

20 In power-residue calculating unit 1900, signal Ywr is a signal for writing the result of the square calculation into Y register 103. Signal Xwr is a signal for writing the multiplication result into X register 106 when  $e[j] = 1$  and into K register 132 when  $e[j] = 0$ . Signal Xrd is a signal for reading a multiplier from X register 106 when  $e[j] = 1$  and from K register 132 when  $e[j] = 0$ .

25 With the configuration as described above, in the processing of power-residue calculating unit 1900, the value in Y register 103 always varies only in the square calculation, irrespective of  $e[j]$ .

Where  $e[j] = 1$  with a multiplier of X, X always varies. Where  $e[j] = 0$  with a multiplier of K, K always varies as well.

30 Therefore the value of the multiplier is different for each iteration so that the correlation of the consumed current in execution of multiplication with  $e[j]$  can be further reduced.

Note that the left binary method is used as follows.

In the case of the left binary method, the iterative portion is as

follows:

```
for j = 1023 to 0
  Y = Y×Y/R mod N
  5    if e[j] = 1  then Y = X×Y/R mod N else J = X×J/R mod N
  end for
```

10 where J is a value stored in a register provided as a dummy of a  
multiplicand. The effect is similar to that in the right binary method. The  
diagram and description showing the configuration thereof will be omitted  
as it can be realized by providing a dummy register J as described above in  
Fig. 10.

15 It is noted that although in the first to eighth embodiments the  
configurations of power-residue calculating units corresponding to the  
respective aims are respectively described above, it is possible to implement  
a power-residue calculating unit capable of handling two or more aims at the  
same time. The present invention can be applied to either the right binary  
method or the left binary method.

20 Although the present invention has been described and illustrated in  
detail, it is clearly understood that the same is by way of illustration and  
example only and is not to be taken by way of limitation, the spirit and scope  
of the present invention being limited only by the terms of the appended  
claims.